



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/919,185	07/30/2001	Edward B. Boden	END920010019US1	2635
7590	11/09/2004		EXAMINER	
IBM Corporation Intellectual Property Law (Dept. 917, Bldg. 006-1) 3605 Highway 52 North Rochester, MN 55901-7829			LESNIEWSKI, VICTOR D	
			ART UNIT	PAPER NUMBER
			2155	

DATE MAILED: 11/09/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/919,185	BODEN, EDWARD B.
	Examiner Victor Lesniewski	Art Unit 2155

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) Responsive to communication(s) filed on 30 July 2001.
 2a) This action is FINAL. 2b) This action is non-final.
 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) Claim(s) 1-53 is/are pending in the application.
 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
 5) Claim(s) _____ is/are allowed.
 6) Claim(s) 1-53 is/are rejected.
 7) Claim(s) _____ is/are objected to.
 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) The specification is objected to by the Examiner.
 10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) Notice of References Cited (PTO-892)
 2) Notice of Draftsperson's Patent Drawing Review (PTO-948)
 3) Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
 Paper No(s)/Mail Date 7/30/2001.

4) Interview Summary (PTO-413)
 Paper No(s)/Mail Date _____.
 5) Notice of Informal Patent Application (PTO-152)
 6) Other: _____.

DETAILED ACTION

1. This application has been examined.
2. Claims 1-53 are now pending.

Information Disclosure Statement

3. The IDS filed on 7/30/2001 has been considered.

Claim Rejections - 35 USC § 101

4. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

5. Claims 44-48 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Claims 44-48 recite descriptive material that may or may not be an embodiment of a computer system or embodied on a computer readable medium so as to be executable. Here, a program storage device readable by a machine does not suffice as computer readable or a computer program product and does not constitute eligible subject matter for patentability. See MPEP 2106.IV.B.1(a).

6. For the purpose of applying prior art it will be assumed that claims 44-48 recite a computer readable medium.

Claim Rejections - 35 USC § 102

7. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

8. Claims 1-14, 16-27, 29-32, 34, and 36-53 are rejected under 35 U.S.C. 102(b) as being clearly anticipated by Fiveash et al. (U.S. Patent Number 6,076,168), hereinafter referred to as Fiveash.

9. Fiveash has disclosed:

- <Claim 1>

A method for control and management of communication traffic, comprising the steps of: expressing access rules as filters referencing system kernel data (column 3, lines 21-25); for outbound processing, determining source application indicia (column 3, lines 8-20); for inbound packet processing, executing a look-ahead function to determine target application indicia (column 2, line 61 through column 3, line 7); and responsive to said source or target application indicia, executing filter processing (column 2, lines 13-14).

- <Claim 2>

The method of claim 1, further comprising the steps of executing said determining and executing steps within a kernel filtering function upon encountering a filter selector field referencing kernel data not included in said packet (column 4, line 64 through column 5, line 22).

- <Claim 3>

The method of claim 1, said filter processing including the steps of: determining a task or thread identifier; based on said task or thread identifier, determining a process or job

identifier (column 1, lines 20-26); and based on said process or job identifier, determining job or process attributes for filter processing (column 5, lines 1-22).

- <Claim 4>

The method of claim 1, said filter processing including the steps of: determining a user identifier (column 1, lines 20-26); and based on said user identifier, determining user attributes for filter processing (column 5, lines 1-22).

- <Claim 5>

The method of claim 3, further comprising the step of determining from said task identifier a work control block containing said process or job identifier (column 1, lines 20-26, as a block of TCP/IP or UDP data).

- <Claim 6>

The method of claim 1, further comprising the steps for inbound processing of: passing an inbound packet to a sockets layer to identify said target application (figure 2, item 29).

- <Claim 7>

The method of claim 6, further comprising the step of marking said inbound packet as not deliverable before passing it to said sockets layer (figure 2, item 28).

- <Claim 8>

The method of claim 1, further comprising the steps of: delivering to said filters infrastructure access rules for defining security context (column 4, lines 36-39).

- <Claim 9>

The method of claim 8, said infrastructure including logging, auditing, and filter rule load controls (column 4, line 64 through column 5, line 22).

- <Claim 10>

A method for control and management of aspects of communication traffic within filtering, comprising the steps of: receiving IP packet data into a TCP/IP protocol stack executing within a system kernel (figure 1, item 110) executing filtering code within said system kernel with respect to non-IP packet data accessed within said system kernel outside of said TCP/IP protocol stack (figure 1, item 120).

- <Claim 11>

The method of claim 10, said non-IP packet data including context data regarding said IP packet (column 3, lines 21-32).

- <Claim 12>

The method of claim 10, said non-IP packet data including data specific to a task generating said non-IP packet data (column 1, lines 20-31).

- <Claim 13>

The method of claim 10, said non-IP packet data including data specific to a task that will receive said IP packet (column 1, lines 20-31).

- <Claim 14>

The method of claim 11, said context data including packet arrival interface indicia (column 3, lines 26-32).

- <Claim 16>

The method of claim 10, further comprising the steps of: establishing a tunnel between two IP address limiting traffic to applications bound to ports at each end of said tunnel;

said filtering code accessing filtering attributes further limiting traffic selectively to job indicia (column 3, lines 64-67).

- <Claim 17>

The method of claim 10, further comprising the steps of: establishing a tunnel between two IP address limiting traffic to applications bound to ports at each end of said tunnel; and filtering code accessing filtering attributes further limiting traffic selectively to user identification indicia (column 3, lines 64-67).

- <Claim 18>

A method for centralizing system-wide communication management and control within filter rules, comprising the steps of: providing filter statements syntax for accepting parameters in the form of a selector, each selector specifying selector field, operator, and a set of values (figure 4); and said selector referencing data that does not exist in IP packets (column 3, lines 21-32).

- <Claim 19>

The method of claim 18, said parameters selectively including userid, user profile, user class, user group, user group authority, user special authority, job name, process name, job group, job class, job priority, other job or process attributes, and date & time (column 1, lines 20-26).

- <Claim 20>

The method of claim 18, said filters statements being provided within a user interface to said system (column 3, lines 57-58).

- <Claim 21>

The method of claim 18, further comprising the steps of: establishing a tunnel between two IP address limiting traffic to applications bound to ports at each end of said tunnel; said filtering code accessing filtering attributes further limiting traffic selectively to job indicia (column 3, lines 64-67); and operating said filtering code within a kernel filtering function upon encountering a filter selector field referencing kernel data not included in said traffic (column 4, line 64 through column 5, line 22).

- <Claim 22>

A method for traversing a portion only of a protocol stack to disallow selective IP packet traffic, comprising the steps of: receiving a packet in the kernel of the operating system of a first node from an application, said kernel including filter processor (column 3, lines 21-25); for inbound packet processing to a first node from a second node, executing a look-ahead function in the system kernel of said first node to determining a target application (column 2, line 61 through column 3, line 7); for both said inbound packet processing, and for outbound packet processing from said first node to said second node, executing within said kernel the steps of processing said packet by determining a task ID (column 1, lines 20-26); responsive to said task ID, determining a corresponding work control block (column 1, lines 20-26, as a block of TCP/IP or UDP data); determining a user process or job identifier from said work control block (column 1, lines 20-26); from the user process or job identifier selectively determining attributes for said user process or job (column 5, lines 1-22); and passing said attributes to said filter processor for managing and controlling communication traffic (column 4, lines 36-39).

- <Claim 23>

A method for expressing access rules as filters, comprising the steps of: providing a filter statements syntax for accepting parameters in the form of a selector, each selector specifying selector field, operator, and a set of values (figure 4); and said selector referencing data that does not exist in IP packets for controlling access to an application (column 3, lines 21-32).

- <Claim 24>

A method for managing and controlling communication traffic by centralizing access rules in filters executing within and referencing data available in system kernels, comprising the steps for outbound packet processing from a first node to a second node of: receiving said packet in the kernel of the operating system of said first node from an application or process at said first node (column 3, lines 21-25); processing said packet by determining a task ID (column 1, lines 20-26); responsive to said task ID, determining a corresponding work control block (column 1, lines 20-26, as a block of TCP/IP or UDP data); responsive to said work control block, determining a process or job identifier (column 1, lines 20-26); responsive to said process job identifier, determining job or process attributes (column 5, lines 1-22).

- <Claim 25>

The method of claim 24, further comprising the steps for inbound packet processing from said second node to said first node of: initially operating said kernel at said first node to determine a target application for said packet at said first node (column 2, line 61 through column 3, line 7).

- <Claim 26>

The method of claim 25, said initially operating step comprising executing a look-ahead function (column 2, line 61 through column 3, line 7).

- <Claim 27>

The method of claim 26, said look-ahead function including the steps of operating a filter function to request of a sockets layer the identity of an application to which said sockets layer would pass said packet (column 1, lines 20-26, especially “routing characteristics”).

- <Claim 29>

A method for managing and controlling communication traffic by centralizing the access rules, comprising the steps for outbound packet processing from a first node to a second node of: receiving said packet in the kernel of the operating system of said first node from an application or process at said first node, said kernel including a filter processor (column 3, lines 21-25); processing said packet by determining a task ID (column 1, lines 20-26); responsive to said task ID, determining a corresponding work control block (column 1, lines 20-26, as a block of TCP/IP or UDP data); determining a user ID control block from said work control block (column 1, lines 20-26); from the user ID control block determining attributes for said user (column 5, lines 1-22); and passing said attributes to said filter processor for managing and controlling communication traffic (column 4, lines 36-39).

- <Claim 30>

The method of claim 29, further comprising the steps for inbound packet processing from said second node to said first node of: initially operating said kernel at said first node to

determine a target application for said packet at said first node (column 2, line 61 through column 3, line 7).

- <Claim 31>

The method of claim 30, said initially operating step comprising executing a look-ahead function (column 2, line 61 through column 3, line 7).

- <Claim 32>

The method of claim 31, said look-ahead function including the steps of operating a filter function to request of a sockets layer the identity of an application to which said sockets layer would pass said packet (column 1, lines 20-26, especially “routing characteristics”).

- <Claim 34>

A method for control and management of communication traffic with respect to a system node, comprising the steps of: receiving at said system node an inbound packet; and executing within a protocol stack of the system kernel of said system node a filtering function identifying for said inbound packet a filter referencing non-packet data (column 2, line 61 through column 3, line 7); and responsive to said filter, executing a look-ahead function for identifying a target application for said inbound packet (column 2, line 61 through column 3, line 7).

- <Claim 36>

System for control and management of communication traffic, comprising: a system kernel including a filter function (figure 1, item 120) and stack data (figure 1, item 110); said filter function including a filter selectively referencing said stack data for expressing access rules (column 3, lines 21-25); said filter function being responsive to receipt of an

outbound packet determining a source application (column 3, lines 8-20); said filter function being responsive to receipt of an inbound packet processing for executing a look-ahead function to determine a target application (column 2, line 61 through column 3, line 7); and said filter function being responsive to said source or target application for executing filter processing (column 2, lines 13-14).

- <Claim 37>

A system for control and management of aspects of communication traffic within filtering, comprising: a system kernel; a protocol stack executing within said system kernel for receiving IP packet data (figure 1, item 110); and filtering code within said system kernel operable with respect to non-IP packet data accessed within said system kernel outside of said protocol stack for controlling and managing said aspects of communication traffic (figure 1, item 120).

- <Claim 38>

A system for centralizing system-wide communication management and control within filter rules, comprising: filter statements having a syntax for accepting parameters in the form of a selector, each selector specifying selector field, operator, and a set of values (figure 4); and said selector referencing data that does not exist in IP packets (column 3, lines 21-32).

- <Claim 39>

A system for traversing a portion only of a protocol stack to disallow selective IP packet traffic, comprising: a system kernel; a filter processor executing within said system kernel (column 3, lines 21-25); said filter processor responsive to an inbound packet for

executing a look-ahead function for determining a target application (column 2, line 61 through column 3, line 7); said filter processor responsive to both inbound and outbound packets for processing said packet by determining a task ID (column 1, lines 20-26); responsive to said task ID, determining a corresponding work control block (column 1, lines 20-26, as a block of TCP/IP or UDP data); determining a user ID, process or job identifier from said work control block (column 1, lines 20-26); from the user ID, process or job identifier selectively determining attributes for said user process or job (column 5, lines 1-22); and passing said attributes to said filter processor for managing and controlling communication traffic (column 4, lines 36-39).

- <Claim 40>

A system for expressing access rules as filters, comprising: a filter statements accepting parameters in the form of a selector, each selector specifying selector field, operator, and a set of values (figure 4); and said selector referencing data that does not exist in IP packets controlling access to an application (column 3, lines 21-32).

- <Claim 41>

A system for managing and controlling communication traffic by centralizing access rules in filters executing within and referencing data available in system kernels, comprising: code for receiving a packet in the kernel of the operating system of a first node from an application or process at said first node (column 3, lines 21-25); code for processing said packet by determining a task ID (column 1, lines 1-20); code responsive to said task ID for determining a corresponding work control block (column 1, lines 20-26, as a block of TCP/IP or UDP data); code responsive to said work control block for

determining a process or job identifier (column 1, lines 1-20); and code responsive to said process or job identifier for determining job or process attributes (column 5, lines 1-22).

- <Claim 42>

A system for managing and controlling communication traffic by centralizing access rules, comprising: a first system node; a second system node (column 2, lines 10-12); a kernel of the operating system of said first system node including a kernel filter processor (column 3, lines 21-25); said kernel for receiving from an application or process at said first system node a packet for communication to said second system node (column 3, lines 8-20); said kernel further for processing said packet by determining a task ID; responsive to said task ID, determining a corresponding work control block; determining a user ID control block from said work control block (column 1, lines 20-26); from the user ID control block determining attributes for said user (column 5, lines 1-22); and passing said attributes to said system kernel filter processor for managing and controlling communication traffic (column 4, lines 36-39).

- <Claim 43>

A system for control and management of communication traffic with respect to a system node, comprising: a filtering function executing within a protocol stack of the system kernel of said system node identifying for an inbound packet a filter referencing non-packet data (column 3, lines 21-25 and figure 1, item 120); and a look-ahead function responsive to said filter for identifying a target application for said inbound packet (column 2, line 61 through column 3, line 7).

- <Claims 44 and 49>

A computer program product or computer program element for control and management of communication traffic according to the steps comprising: expressing access rules as filters referencing system kernel data (column 3, lines 21-25); for outbound processing, determining a source application (column 3, lines 8-20); for inbound packet processing, executing a look-ahead function to determine a target application (column 2, line 61 through column 3, line 7); and responsive to said source or target application, executing filter processing (column 2, lines 13-14).

- <Claims 45 and 50>

A computer program product or computer program element for control and management of aspects of communication traffic within filtering according to steps comprising: receiving IP packet data into a TCP/IP protocol stack executing within a system kernel (figure 1, item 110) executing filtering code within said system kernel with respect to non-IP packet data accessed within said system kernel outside of said TCP/IP protocol stack (figure 1, item 120).

- <Claims 46 and 51>

A computer program product or computer program element for centralizing system-wide communication management and control within filter rules according to method steps comprising: providing filter statements syntax for accepting parameters in the form of a selector, each selector specifying selector field, operator, and a set of values (figure 4); and said selector referencing data that does not exist in IP packets (column 3, lines 21-32).

- <Claims 47 and 52>

A computer program product or computer program element for managing and controlling communication traffic by centralizing access rules in filters executing within and referencing data available in system kernels according to method steps comprising: receiving said packet in the kernel of the operating system of said first node from an application or process at said first node (figure 1, item 110); processing said packet by determining a task ID (column 1, lines 20-26); responsive to said task ID, determining a corresponding work control block (column 1, lines 20-26, as a block of TCP/IP or UDP data); responsive to said work control block, determining a process or job identifier (column 1, lines 20-26); responsive to said process or job identifier, determining job or process attributes (column 5, lines 1-22).

- <Claims 48 and 53>

The computer program product or element of claim 52, said method steps further comprising for inbound packet processing from said second node to said first node: initially operating said kernel at said first node to determine a target application for said packet at said first node (column 2, line 61 through column 3, line 7).

Since all the limitations of the invention as set forth in claims 1-14, 16-27, 29-32, 34, and 36-53 were disclosed by Fiveash, claims 1-14, 16-27, 29-32, 34, and 36-53 are rejected.

Claim Rejections - 35 USC § 103

10. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

11. Claims 15, 28, 33, and 35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Fiveash, as applied above, in view of Cunningham et al. (U.S. Patent Number 6,219,786), hereinafter referred to as Cunningham.

12. Fiveash disclosed a method for securing data traffic between host systems that uses a filter having rules associated with a defined tunnel. In an analogous art, Cunningham disclosed a rules base for monitoring network traffic and assembling data packets. Both systems extensively describe a set of rules used in packet transfer to permit or deny the transmission of data.

13. Concerning claim 15, Fiveash does not explicitly state the use of packet arrival time as a property for tracking data transmission in his system. However, Cunningham states the use of a time-of-day property in his rules base. Since the inventions solve the same problem of filtering data through access rules, it would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the system of Fiveash by adding the ability to utilize a time-of-day property as provided by Cunningham. Here, the combination satisfies the need for a more user-friendly and easier to administer rules base in packet filtering systems. See Cunningham, column 3, lines 16-18.

14. Concerning claims 28, 33, and 35, Fiveash does not explicitly state forwarding a non-deliverable packet to a sockets layer. Although he speaks extensively about processing packets and identifying a packet destination, his system seems to discard non-deliverable packets and it is unclear as to whether they may be passed to a sockets layer after being processed as non-deliverable. However, Cunningham's system allows a process where non-deliverable packets

may be analyzed again by checking application layer information, even if they have been previously determined to be non-deliverable. Since the inventions solve the same problem of filtering data through access rules, it would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the system of Fiveash by adding the ability to check non-deliverable packets at a higher layer as provided by Cunningham. Again, the combination satisfies the need for a more user-friendly and easier to administer rules base in packet filtering systems. See Cunningham, column 3, lines 16-18.

15. Thereby, the combination of Fiveash and Cunningham discloses:

- <Claim 15>

The method of claim 11, said context data including packet arrival time-of-day indicia (Cunningham, column 4, lines 40-44).

- <Claim 28>

The method of claim 27, further comprising the step of marking said packet as non-deliverable and thereafter passing said packet to said sockets layer to identify said application (Cunningham, column 10, lines 9-20).

- <Claim 33>

The method of claim 32, further comprising the step of marking said packet as non-deliverable and thereafter passing said packet to said sockets layer to identify said application (Cunningham, column 10, lines 9-20).

- <Claim 35>

The look-ahead function of the method of claim 34 further comprising the steps of: passing to a transport layer function identified by an IP header a packet marked non-

deliverable for determining which user-level process or job is to receive said packet (Cunningham, column 10, lines 9-20); receiving from said transport layer an application layer task identifier said user-level process or job (Fiveash, column 1, lines 20-26 or Cunningham, figure 6, item 82); and thereafter passing said packet marked by said task identifier to said transport layer for delivery to said application layer task (Fiveash, figure 2, item 29).

Since the combination of Fiveash and Cunningham discloses all of the above limitations, claims 15, 28, 33, and 35 are rejected.

Conclusion

16. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Victor Lesniewski whose telephone number is 571-272-3987. The examiner can normally be reached on Monday through Thursday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Hosain Alam can be reached on 571-272-3978. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

VL
Victor Lesniewski
Patent Examiner
Group Art Unit 2155

Hosain Alam
HOSAIN ALAM
SUPERVISORY PATENT EXAMINER